

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Gašper Gartner

**Uvedba davčne blagajne v povezavi s sistemom za podporo
servisa in prodaje vozil**

DIPLOMSKO DELO NA VISOKOŠOLSKEM STROKOVNEM ŠTUDIJU

Ljubljana, 2016

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Gašper Gartner

**Uvedba davčne blagajne v povezavi s sistemom za podporo
servisa in prodaje vozil**

DIPLOMSKO DELO NA VISOKOŠOLSKEM STROKOVNEM ŠTUDIJU

MENTOR: prof. dr. Viljan Mahnič

Ljubljana, 2016

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Uvedba davčne blagajne v povezavi s sistemom za podporo servisa in prodaje vozil

Tematika naloge:

Poslovanje servisne in prodajne mreže enega izmed slovenskih uvoznikov avtomobilov je podprto s sistemom CROSS, ki ga je bilo treba prilagoditi zahtevam Zakona o davčnem potrjevanju računov. Proučite tehnične zahteve davčnega potrjevanja računov v Republiki Sloveniji in postopek izdelave računov v sistemu CROSS. Na podlagi tega izdelajte rešitev, ki prestreže vsak račun, izdelan v sistemu CROSS, ga opremi s potrebnimi podatki in sproži njegovo potrditev pri Finančni upravi Republike Slovenije. Predstavite podatkovni model rešitve ter opišite uporabljena programska orodja in storitve, ki so na voljo končnemu uporabniku.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Gašper Gartner, vpisna številka 63990056, avtor zaključnega dela z naslovom:

Uvedba davčne blagajne v povezavi s sistemom za podporo servisa in prodaje (angl. Implementing fiscal cash register in combination with dealership management system)

IZJAVLJAM

1. da sem pisno zaključno delo študija izdelal samostojno pod mentorstvom prof. dr. Viljana Mahniča;
2. da je tiskana oblika pisnega zaključnega dela študija istovetna elektronski obliki pisnega zaključnega dela študija;
3. da sem pridobil vsa potrebna dovoljenja za uporabo podatkov in avtorskih del v pisnem zaključnem delu študija in jih v pisnem zaključnem delu študija jasno označil;
4. da sem pri pripravi pisnega zaključnega dela študija ravnal v skladu z etičnimi načeli in, kjer je to potrebno, za raziskavo pridobil soglasje etične komisije;
5. soglašam, da se elektronska oblika pisnega zaključnega dela študija uporabi za preverjanje podobnosti vsebine z drugimi deli s programsko opremo za preverjanje podobnosti vsebine, ki je povezana s študijskim informacijskim sistemom članice;
6. da na UL neodplačno, neizključno, prostorsko in časovno neomejeno prenašam pravico shranitve avtorskega dela v elektronski obliki, pravico reproduciranja ter pravico dajanja pisnega zaključnega dela študija na voljo javnosti na svetovnem spletu preko Repozitorija UL;
7. dovoljujem objavo svojih osebnih podatkov, ki so navedeni v pisnem zaključnem delu študija in tej izjavi, skupaj z objavo pisnega zaključnega dela študija.

V Ljubljani, 20. julij 2016

Podpis študenta/-ke:

Zahvaljujem se prof. dr. Viljanu Mahniču za vso pomoč in izredno hitro odzivnost tudi v času počitnic.

Rad bi se zahvalil tudi družini za vso podporo pri zaključevanju študija.

Hvala tudi vsem ostalim za spodbudo.

Kazalo

Povzetek

Abstract

Poglavje 1	Uvod	1
Poglavje 2	Davčne blagajne v Republiki Sloveniji	3
2.1	Digitalna potrdila	3
2.2	Vrste sporočil	3
2.2.1	Računi	3
2.2.2	Podatki o poslovnem prostoru	5
2.2.3	Sporočilo tipa »ECHO«	5
2.3	Elektronski podpis sporočil v XML obliki	5
2.4	Zaščitna oznaka izdajatelja računa (ZOI)	5
2.5	Izpis QR kode, PDF417 kode in CODE 128	6
Poglavje 3	Predstavitev problema in zahtev naročnika	7
3.1	Sistem za podporo in servisiranje vozil	7
3.2	Zahteve naročnika	7
3.3	Izvozne datoteke sistema CROSS	7
3.3.1	Podatki, ki jih je zagotovil naročnik	8
3.3.2	E-računi	9
Poglavje 4	Izvedba	11
4.1	Centraliziran aplikacijski in podatkovni strežnik	11
4.2	Podatkovni model	12
4.3	Uporabljena programska orodja	14
4.3.1	Razvojno okolje Delphi	14
4.3.2	Microsoft SQL server	16

4.4	Storitve aplikacijskega nivoja	16
4.4.1	iDataBroker	16
4.4.2	iBroadcast	17
4.4.3	iAgent.....	18
4.4.4	iNadzor.....	19
4.5	Odjemalec	20
4.5.1	Uporabniki	21
4.5.2	Nastavitve davčnega potrjevanja	21
4.5.3	Pregled računov.....	23
4.5.4	Pregled in urejanje e-računov	23
Poglavje 5	Sklepne ugotovitve	25
Literatura.....		27

Seznam uporabljenih kratic

kratica	angleško	slovensko
DLL	Dynamic-link library	Dinamično povezana knjižnica
EOR	The unique identification invoice mark	Enkratna identifikacijska oznaka računa
FURS	Financial administration of Republic Slovenia	Finančna uprava Republike Slovenije
ISFU	Information system of the financial administration	Informacijski sistem finančne uprave
JSON	JavaScript Object Notation	Format za izmenjavo podatkov, temelječ na programskem jeziku JavaScript
SFTP	Secure File Transfer Protocol	Varni protokol za prenos datotek
TLS	Transport Layer Security	Varnost prenosnega sloja (protokoli, ki omogočajo šifrirano povezavo med strežnikom in odjemalcem)
VCL	Visual Component Library	Knjižnica slikovnih komponent
XML	Extensible Markup Language	Razširljivi označevalni jezik
XSD	XML Schema Definition	XML definicija sheme
ZOI	The protective mark of the invoice issuer	Zaščitna oznaka izdajatelja računa

Povzetek

Naslov: Uvedba davčne blagajne v povezavi s sistemom za podporo servisa in prodaje vozil

V diplomski nalogi je predstavljena rešitev uvedbe davčne blagajne v sistemu CROSS, kjer direktna implementacija ni bila izvedljiva. CROSS je sistem za podporo servisiranja in prodaje vozil, ki ga uporablja mreža trgovcev po Sloveniji in tujini. Z izmenjavo XML datotek se prestreže izdani račun, opremi s potrebnimi podatki in potrdi na Finančno upravo Republike Slovenije. Sistem temelji na centralnem aplikacijskem strežniku, na katerega se povezuje preko 200 odjemalcev. Centralni strežnik se povezuje še s podatkovno bazo, strežnikom finančne uprave in strežnikom podjetja SETCCE, ki skrbi za pošiljanje e-računov. Predstavljene so tehnične zahteve davčnega potrjevanja v Republiki Sloveniji, relacijski podatkovni model in glavni orodji pri izvedbi (programsko okolje Delphi in Microsoft SQL Server). Podrobno so predstavljene vse potrebne storitve, ki tečejo na centralnem aplikacijskem strežniku in aplikacija odjemalca. Odjemalec je Windows aplikacija, katere glavna naloga je, da trgovcu izpiše davčno potrjen račun, ki se ga izstavi stranki. Omogoča še vrsto drugih nastavitev, ki so potrebne za davčno potrjevanje in vsebuje tudi orodje za pripravo e-računov. Na koncu je opisano še delovanje v produkciji in dopolnitve, ki se bodo izvajale naknadno.

Ključne besede: diplomska naloga, davčno potrjevanje, davčna blagajna, Delphi, Microsoft SQL server

Abstract

Title: Implementing fiscal cash register in combination with dealership management system

The thesis presents implementation of fiscal cash registers on the CROSS system, where direct implementation was not feasible. CROSS is a dealership management system used by a network of retailers in Slovenia and over the world. Through the exchange of XML files, invoice data is captured, issued with the necessary information and fiscal verified by Financial Administration of the Republic of Slovenia. The system is based on a central application server with over 200 client connections. The central server also connects to a database server, Financial Administration server and SETCCE server to handle the sending of e-invoices. Presented are the technical specifications for fiscal verification of invoices in the Republic of Slovenia, the relational data model and the main tools used for implementation (programming environment Delphi and Microsoft SQL Server). There is a detailed presentation of all services that run on a central application server and client application. The client is a Windows based application. The main task of client is to print fiscal verified invoice at the dealer, that is issued to a customer. It allows a variety of other settings that are required for tax verification and also contains a tool for e-invoices. At the end, a description of system operation and possible further upgrades is provided.

Keywords: thesis, fiscal cash register, fiscal verification, Delphi, Microsoft SQL server

Poglavje 1 Uvod

Z 2. januarjem 2016 je stopil v veljavo Zakon o davčnem potrjevanju računov. Veliko večjih podjetij je s tem imelo težave pri uvedbi davčne blagajne na obstoječem informacijskem sistemu, še posebej v primeru kjer sistem ni domače izdelave. Eden od uvoznikov avtomobilov svojim trgovcem ponuja celovito rešitev za vodenje servisiranja in prodaje vozil – sistem CROSS. Žal vzdrževalci sistema niso ponudili ustrezne rešitve za podporo davčnim blagajnam, kot jo zahteva slovenska zakonodaja. Uvoznik se je tako srečal s problemom, ki ga je bilo potrebno rešiti z drugo zunanjo rešitvijo. V podjetju iPlus, d. o. o., smo ponudili ustrezno rešitev, ki jo je uvoznik tudi izbral. Rešitev temelji na podlagi izmenjave podatkov o izdanem računu. Sistem CROSS je centraliziran, tako da se vsaka akcija, ki jo izvede trgovec, dogaja na centralnem strežniku. To je bistveno olajšalo izvedbo, saj izmenjava podatkov teče samo na enem mestu za vse trgovce. Tudi naša rešitev temelji na centraliziranem strežniku – vsak trgovec si je na svojih delovnih postajah moral namestiti še odjemalca, ki v prvi vrsti omogoča hiter izpis računa z vsemi zahtevami davčnih blagajn. Poleg tega smo ugodili še pri nekaterih dodatnih zahtevah izpisa in pošiljanja e-računov.

V poglavju Davčne blagajne v Republiki Sloveniji si bomo ogledali, kaj zapoveduje zakonodaja in podrobneje pogledali tehnične zahteve davčnega potrjevanja. V poglavju 3 je predstavljen problem in zahteve naročnika, kjer bomo pogledali, kaj je sistem CROSS, v kakšni obliki lahko prejmemo podatke o izdanih računih, kako lahko izboljšamo obstoječo rešitev pošiljanja e-računov in dodatne zahteve naročnika. V poglavju Izvedba si bomo ogledali trinivojsko arhitekturo, ki temelji na odjemalcu in strežnikih. Podrobneje si bomo ogledali relacijski podatkovni model, uporabljena programska orodja, storitve, ki tečejo na aplikacijskem strežniku, in aplikacijo odjemalca. Na koncu sledi zaključek s sklepnimi ugotovitvami.

Poglavje 2 Davčne blagajne v Republiki Sloveniji

15. julija 2015 je bil sprejet Zakon o davčnem potrjevanju računov, ki je stopil v veljavo 2. januarja 2016. Vsak zavezanec mora ob dobavi blaga ali storitve za plačilo z gotovino izdati račun preko ustrezne elektronske naprave, ki izpolnjuje predpisane pogoje in omogoča izvedbo postopka potrjevanja računov.

Elektronska naprava je preko spleta povezana s centralnim informacijskim sistemom Finančne uprave RS, ki potrjuje in shranjuje podatke o računih v realnem času. Sistem omogoča sledljivost in učinkovitost nadzora nad izdanimi računi ter bo omejil sivo ekonomijo [1].

V naslednjih poglavjih si bomo ogledali povzetek tehnične dokumentacije, kot je objavljena na internetni strani FURS [2].

2.1 Digitalna potrdila

Vsak zavezanec mora preko portala eDavki pridobiti namensko digitalno potrdilo, ki se uporablja za avtentikacijo TLS seje in podpisovanje sporočil. Na voljo sta dve potrdili, eno za testno in drugo za produkcijsko okolje.

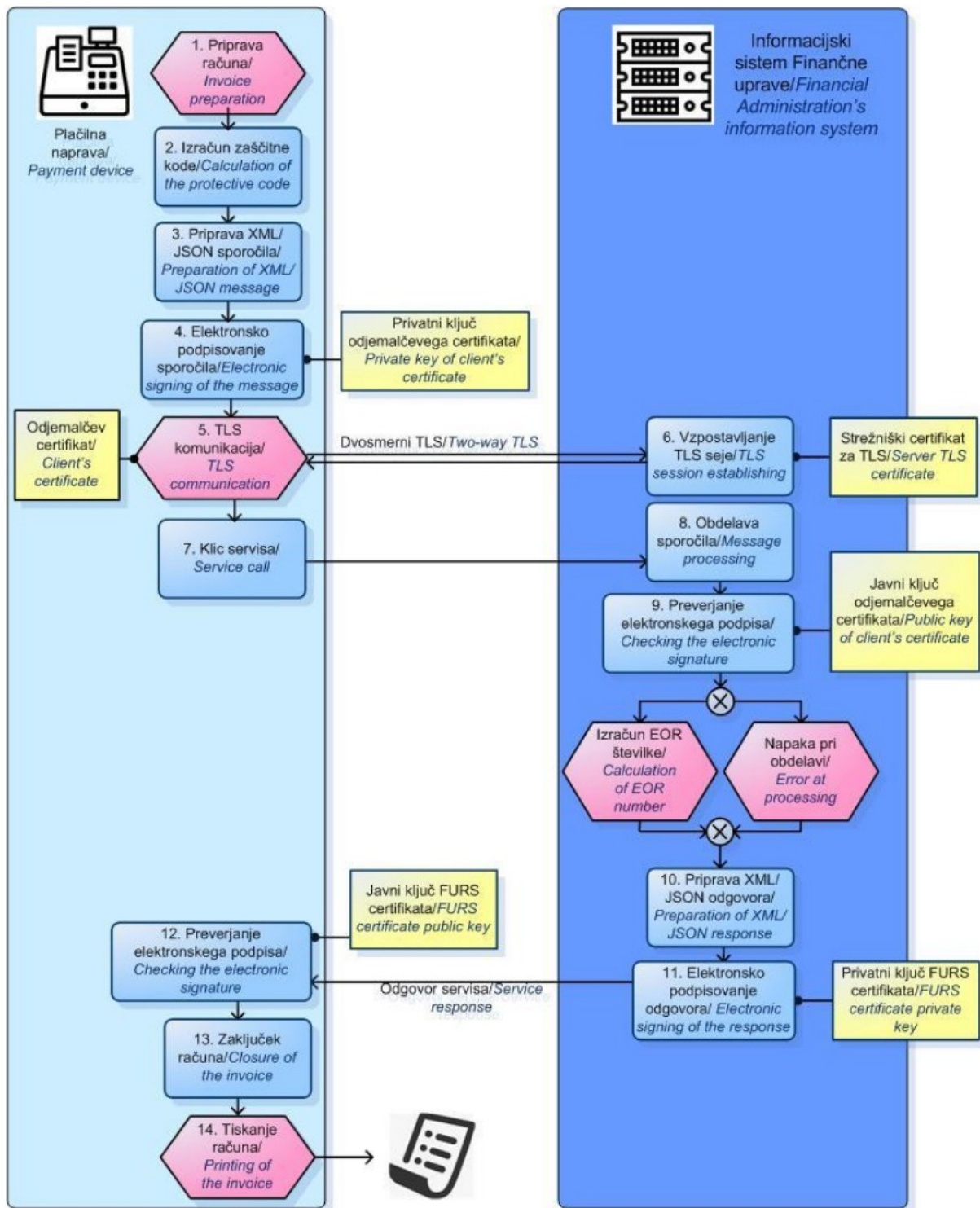
2.2 Vrste sporočil

Davčni zavezanci so dolžni poslati na davčni organ Finančne uprave RS dva tipa sporočil: račun in podatke o poslovnih prostorih. Sporočilo je lahko v XML ali JSON obliki.

2.2.1 Računi

Račun vsebuje vse pomembne podatke, kot so davčna številka, datum in čas izdaje računa, številka računa, skupna vrednost, davčna osnova in DDV, razdeljen po davčnih stopnjah itd. V odgovoru prejmemo vsebinske napake ali pa identifikacijsko oznako (EOR), ki jo moramo izpisati na račun.

Postopek priprave in pošiljanja podatkov o računu je prikazan na sliki 2.1. Ko iz plačilne naprave dobimo podatke o računu, najprej izračunamo zaščitno kodo (ZOI), nato pripravimo XML ali JSON sporočilo, ki ga elektronsko podpišemo.



Slika 2.1: Postopek priprave in pošiljanja podatkov o računu [2]

Sledi vzpostavitev varne TLS povezave z informacijskim sistemom Finančne uprave RS s pomočjo odjemalčevega in strežniškega certifikata. S klicem storitve oddamo predhodno pripravljeno sporočilo, ki se obdela na informacijskem sistemu Finančne uprave RS. Z javnim ključem odjemalčevega certifikata se preveri elektronski podpis. V primeru ko je sporočilo vsebinsko in sintaktično pravilno, se mu izračuna identifikacijska oznaka (EOR), sicer se generira sporočilo o napaki. Sledi priprava XML ali JSON odgovora, ki se ga podpiše s privatnim ključem FURS certifikata. Na strani plačilne naprave to sporočilo prejmemo, preverimo elektronski podpis, zaključimo in izpišemo račun. V primeru neodzivnosti strežnika na FURS imamo možnost izdaje računa brez identifikacijske oznake (EOR), vendar moramo tak račun naknadno potrditi v roku dveh delovnih dni.

2.2.2 Podatki o poslovnem prostoru

Vsak poslovni prostor, od koder se bo izdajalo račune, je potrebno predhodno registrirati. Na FURS je potrebno poslati podatke, kot so davčna številka, oznaka poslovnega prostora, naslov, številka katastrske občine, številka stavbe, ali gre za premični poslovni prostor in drugo. Odgovor dobimo v obliki potrditve ali opisa napake.

2.2.3 Sporočilo tipa »ECHO«

Sporočilo tipa »ECHO« se uporablja za testiranje delovanja povezave ali pravilnosti podpisa in strukture XML sporočila.

2.3 Elektronski podpis sporočil v XML obliki

Vsako sporočilo je potrebno podpisati z digitalnim potrdilom, s tem zagotovimo nespremenjenost vsebine. Podatkov o javnem ključu ni treba pošiljati, ker jih ima ISFU shranjene lokalno, metoda podpisovanja in zgoščevanja pa je SHA256.

2.4 Zaščitna oznaka izdajatelja računa (ZOI)

Zaščitna oznaka izdajatelja računa (ZOI) je alfanumeričen zapis, ki ga mora izračunati izdajatelj računa. Prav tako jo mora izpisati na računu in posredovati na finančno upravo v sporočilu računa. Pri izračunu ZOI se uporabi naslednje podatke: davčna številka, datum izdaje računa, zaporedna številka računa, oznaka poslovnega prostora, oznaka elektronske naprave in znesek računa. Podatke se elektronsko podpiše z RSA-SHA256 in namenskim digitalnim potrdilom, izdanim s strani MJU za namen davčnega potrjevanja. Na teh podatkih se uporabi MD5 zgoščevalno funkcijo in tako dobimo končno vrednost ZOI.

2.5 Izpis QR kode, PDF417 kode in CODE 128

Na izpisu računa mora biti zaščitna oznaka izdajatelja računa (ZOI) navedena v tekstovni obliki (šestnajstiški zapis) in v obliki QR kode, kode PDF 417 ali več črtnih kod tipa Code 128. Vsebina vseh treh kod je ZOI, pretvorjena v desetiški zapis, davčna številka, datum in čas izdaje računa in kontrolni znak. V našem primeru smo se odločili za QR kodo, saj za to ni bilo tehničnih omejitev tiskalnika, ki bi zahtevale uporabo enega od drugih dveh tipov. Berljivost kode se lahko preveri z namensko aplikacijo, ki je na voljo za večino pametnih naprav.

Poglavje 3 Predstavitev problema in zahtev naročnika

V tem poglavju bomo opisali, kaj naša aplikacija omogoča in zakaj je bila za naročnika dobra rešitev.

3.1 Sistem za podporo in servisiranje vozil

Naročnik ima za vse aktivnosti poslovanja centralni sistem CROSS [3], ki ga uporablja mreža trgovcev po celi Sloveniji. Sistem optimizira celotno dobavno verigo; začnši s stranko preko jedra – trgovca, do uvoznika in proizvajalca. Na ta način se potek dela v avtohiši bistveno izboljša. Ne glede na to od kod pride informacija, je za pogovor s stranko na voljo ob pravem času in je ustrezno ažurna.

Sistem uporablja 16 držav po vsem svetu, zaradi tega je žal zelo splošen in nefleksibilen glede prilagajanja posameznemu naročniku. Tu ni šlo samo za problem davčnih blagajn, temveč tudi za samo obliko izpisa izdanega računa in podporo e-računom.

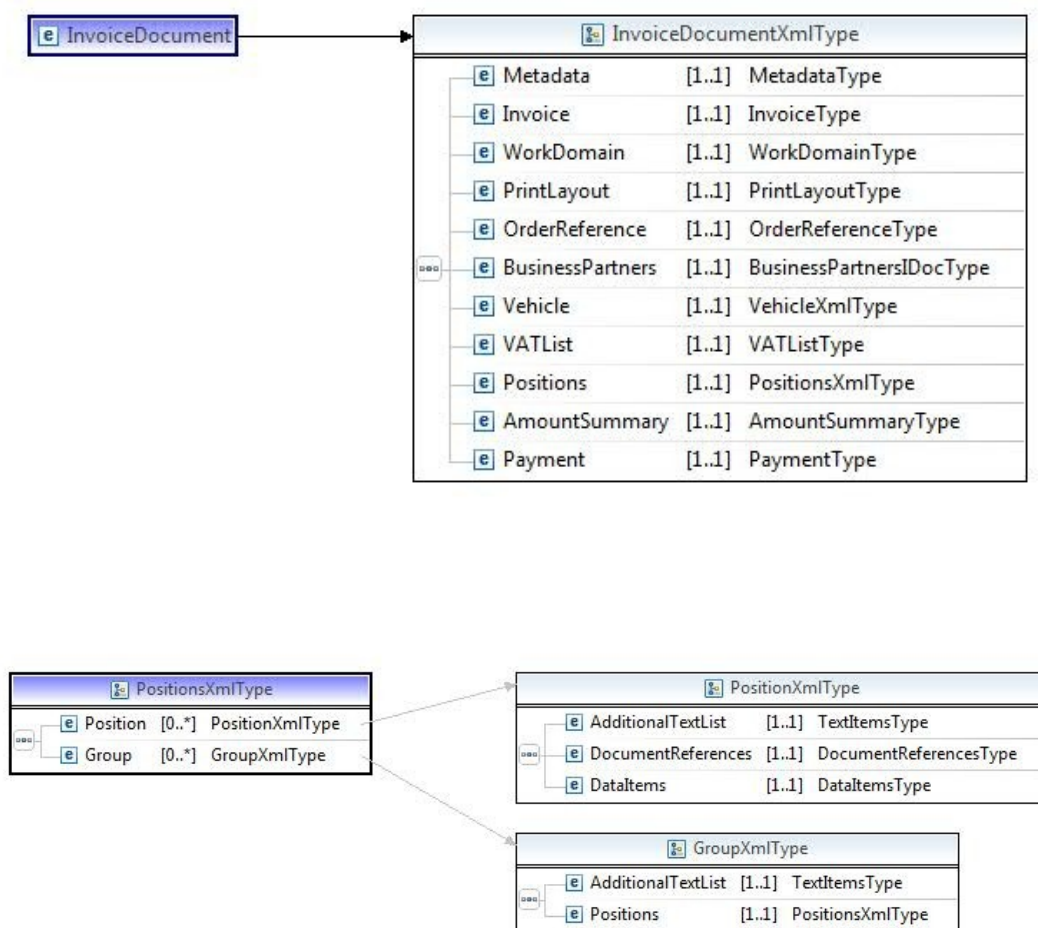
3.2 Zahteve naročnika

Naročnik je želel rešitev, ki bo rešila problem davčnih blagajn pri vseh trgovcih v Sloveniji, sistem mora biti enostaven za namestitev in uporabo ter mora omogočati samodejne posodobitve. Teči mora na centralnem strežniku, ki ga vzdržuje in priskrbi naročnik. Poleg davčnega potrjevanja računov so bili naročniku bistvenega pomena izpisi računa. Naročnik je definiral dva izpisa, enega za prodajo in enega za servis vozil.

3.3 Izvozne datoteke sistema CROSS

Sistem CROSS omogoča avtomatiziran izvoz različnih podatkov, tako je naročnik že imel rešeno pošiljanje e-računov, ki so ga bili primorani implementirati že leto prej kot davčne blagajne. Naša rešitev temelji na istem konceptu. Izvozne datoteke so v XML obliki, za katero obstaja dokumentacija v XSD obliki. V tej shemi so definirani vsi možni elementi, njihovi tipi in relacije med njimi. Del te sheme je predstavljen na sliki 3.1. Shema je obsežna in kompleksna, posebno problematičen je tip s postavkami računa (PositionsXmlType), ki po definiciji preko tipa GroupXMLType lahko raste neskončno v globino. Drugi problem so

nekatera polja, ki so bila prisotna v primerih XML datotek in niso bila opisana v shemi, kar je pomenilo, da shema ni bila v najnovejši različici. Problem manjkajoče dokumentacije je bilo treba reševati na podlagi velikega števila predloženih XML primerov in s sestanki z naročnikom.



Slika 3.1: Del definicije XML sheme izdanega računa

3.3.1 Podatki, ki jih je zagotovil naročnik

Nekatere zakonske zahteve je lahko zagotovil kar naročnik sam in nam je tako olajšal delo. Številčenje računov za davčno potrjevanje mora biti v okviru poslovnega prostora in blagajne, ta dva podatka morata biti tudi navedena pred samo številko računa, ločena s pomišljajem. Naročnik je lahko zagotovil pravilno številčenje že na sistemu CROSS, naša naloga je bila, da

med prvi dve številki dodamo pomišljaje. V številki torej dobimo informacijo, za kateri poslovni prostor in elektronsko napravo gre. Številčenje se znotraj tekočega leta ne sme ponoviti, prav tako številka ne sme vsebovati vodečih ničel. Zakon sicer dopušča dvojno številčenje, tako bi lahko interno številčenje ostalo enako kot prej, v davčni blagajni pa bi se generirala nova številka računa v okviru poslovnega prostora in blagajne, ampak v tem primeru bi naleteli na težavo, ker na strežniku ne bi vedeli, na kateri blagajni je bil izdan račun.

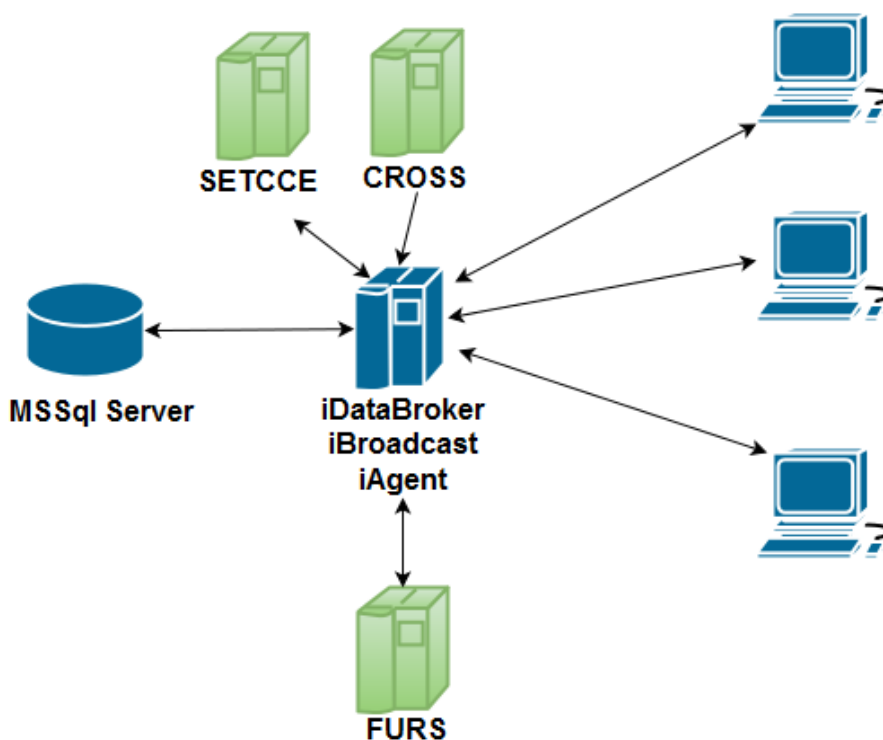
3.3.2 E-računi

Naročnik je že imel rešitev za pošiljanje e-računov s podjetjem SETCCE, d. o. o. Gre za sistem izmenjave XML datoteke računa preko SFTP protokola. Ker je bila priprava in pošiljanje datoteke računa izvedena preveč okorno, je naročnik želel novo rešitev, ki bo omogočala boljši pregled, povratne informacije, dodajanje priponk in izpis računa v XML datoteko.

Poglavje 4 Izvedba

4.1 Centraliziran aplikacijski in podatkovni strežnik

Naša tehnična rešitev temelji na trinivojski arhitekturi: predstaviteni, aplikacijski in podatkovni nivo. Na centralni aplikacijski strežnik se povezujejo uporabniki z različnih lokacij tudi preko internetne povezave. Aplikacijski strežnik komunicira s podatkovnim strežnikom, kjer teče Microsoftov SQL Server. SQL Server nima samo podatkovne vloge, ampak se vpleta tudi v aplikacijski nivo s shranjenimi procedurami. Model je predstavljen na sliki 4.1. Nekaj preglavic nam je povzročala stroga varnostna politika naročnika, kjer je bilo treba več časa za odobritev odprtja podatkovnih vrat v požarnem zidu. Na predstavitenem nivoju teče odjemalec – Windows aplikacija.



Slika 4.1: Model odjemalec strežnik

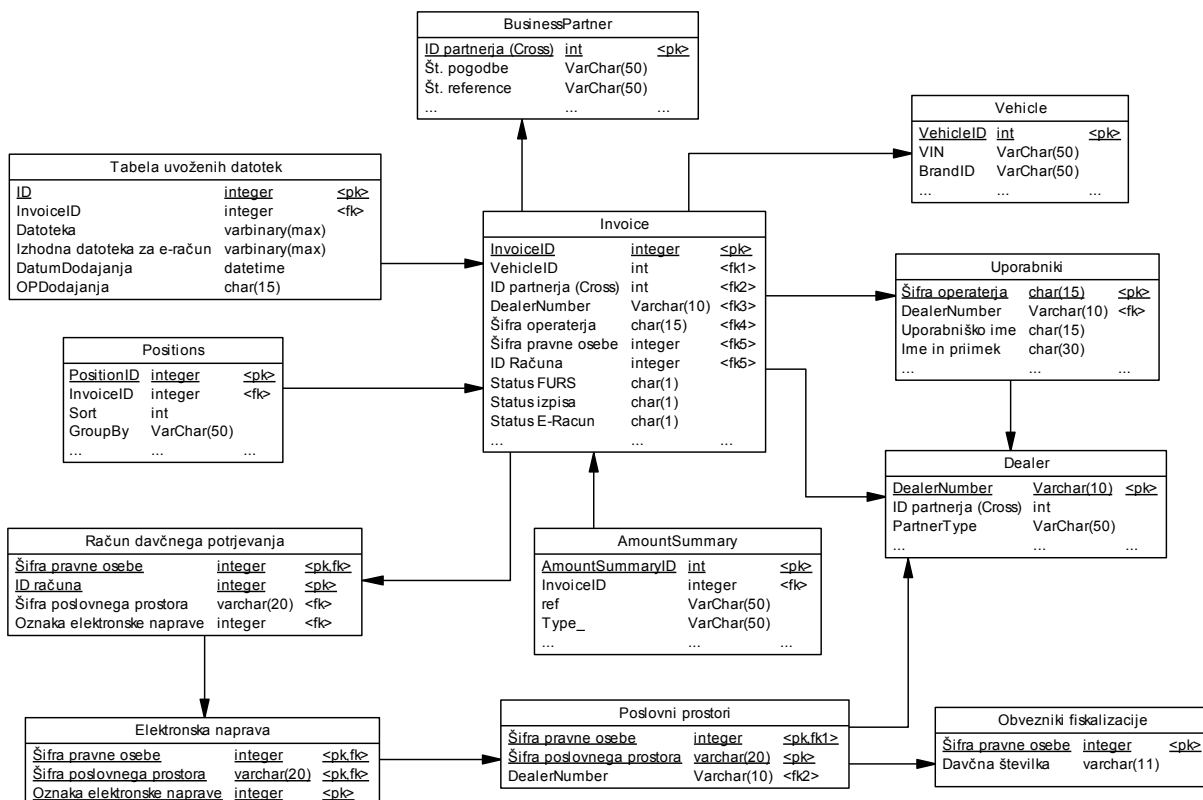
Večina procesov (uvoz podatkov, priprava podatkov, potrjevanje računov) teče na aplikacijskem strežniku ali direktno na podatkovni bazi. Komunikacija poteka preko namenskega servisa iDataBroker, to nam zagotavlja, da na odjemalcu ni potrebna namestitve nobenega gonilnika za podatkovno bazo ali kakšne dodatne programske knjižnice. Vsa

komunikacija poteka preko TCP/IP, je kompresirana in kriptirana. Sistem nam zagotavlja tudi avtomatske posodobitve aplikacije na odjemalcu. Povezava s sistemom CROSS je enosmerna, na aplikacijski strežnik v dogovorjeno skupno mapo prejmemo XML datoteko, ki jo obdela iBroadcast. S strežnikom FURS komunicira samo podatkovni strežnik, ker določeni odjemalci zaradi stroge varnostne politike nimajo prostega internetnega dostopa in bi bilo potrebno odpirati dostop na več mestih, kar bi bilo zamudno. Za dostavo e-računov se uporablja SFTP povezavo na SETCCE strežniku, povezava je dvosmerna, saj tudi mi moramo brati povratno informacijo pošiljanja e-računov.

4.2 Podatkovni model

Naročnik nam je predložil zelo skopo dokumentacijo, ki je zajemala samo XSD definicijo izvozne datoteke. Glede na to definicijo, veliko primerov izvoznih datotek in sestank pri naročniku smo si oblikovali podatkovni model z orodjem Power Designer. Za boljši pregled in drevesno strukturo XSD datotek smo uporabili orodje Eclipse. Poskušali smo ohraniti imena polj in struktur tudi v podatkovnem modelu zaradi lažjega pretvarjanja med XML datoteko in podatkovno bazo. Slika 4.1 prikazuje del relacijskega podatkovnega modela, ki je nastal na podlagi podatkov računa. Tabela Invoice je osnovna tabela in vsebuje osnovne podatke računa. Atribut InvoiceId je primarni ključ tabele, ki je tipa identity, kar pomeni, da se mu vrednost dodeli avtomatsko ob vsakem vnosu. Tabela vsebuje še podatke o vozilu (VehicleId), partnerju (BusinessPartnerId), trgovcu (Dealer), številko računa in še mnogo drugih podatkov, ki so nujno potrebni za izpis in davčno potrditev računa. V tabeli Positions so postavke računa, npr. podatki o posamezni storitvi, materialu ali blagu. V primeru prodaje vozila je v posamezni postavki opis dodatne opreme vozila. Postavka lahko vsebuje ceno, lahko pa tudi ne in ima samo vlogo opisa. Tabela AmountSummary vsebuje rekapitulacijo davka po posameznih davčnih skupinah, v njej so tudi ločeno prikazana predplačila, v primeru da jih je treba odšteti od skupnega zneska.

Vsaka uvožena XML datoteka se shrani v tabelo uvoženih datotek, to nam zagotavlja hiter vpogled v uvožene podatke v primeru, ko je treba zadeve dodatno pojasnjevati ali dopolniti. Služi nam tudi za hrambo izvoznih datotek, ki jih iz našega sistema pošljemo naprej za potrebe e-računa.



Slika 4.2: Poenostavljen podatkovni model na osnovi XML izvozne datoteke

Z vsakim uvozom se ažurirajo tudi tabele Dealer (trgovec), BusinessPartner (poslovni partner), Vehicle (vozilo). V tabeli Dealer so vsi trgovci, ki prodajajo in servisirajo vozila in uporabljajo naš sistem. Z davčnega vidika so to kar poslovni prostori, saj so tako že definirani v sistemu CROSS. V njej hranimo podatke o davčni in matični številki, številki transakcijskega računa in druge podatke. V tabeli BusinessPartner so podatki o prejemniku računa, lahko gre za podatke fizične ali pravne osebe, v prvem primeru imamo ime in priimek, v drugem pa naziv podjetja. Vodimo še naslednje podatke: naslov, davčna številka, številka pogodbe, številka reference, ali gre za vladno organizacijo in ali se pošilja račun preko elektronske pošte. V tabeli Vehicle vodimo podatke o vozilu, kot so znamka, model, barva, oznaka, identifikacija vozila (VIN), registrska številka itd.

Potrebne so bile še standardne tabele, ki jih uporabljamo pri večjih produktih, to so splošni šifranti. V tabeli Uporabniki so končni uporabniki davčne blagajne, v njej hranimo uporabniško ime, ime in priimek osebe, kateremu trgovcu pripada, davčno številko in druge. V tabeli Pooblastila imamo definirana osnovna pooblastila po posameznih uporabniških akcijah in so del tabele Profili, v kateri zajamemo sklop pooblastil, ki jih bo imel nek nivo uporabnikov. S to tabelo smo definirali tri nivoje: administrator, administrator pri trgovcu in uporabnik.

Za potrebe davčnega potrjevanja smo definirali tri glavne šifrantе: Obvezniki fiskalizacije, Poslovni prostori in Elektronske naprave. V tabeli Obvezniki fiskalizacije so podatki pravnih oseb. V njej hranimo podatke o nazivu podjetja, davčni številki, digitalni certifikat, geslo za certifikat in ali je davčni zavezanec ali ne. V tabeli Poslovni prostori hranimo vse zakonsko potrebne podatke za elektronsko registracijo poslovnega prostora, kot so naziv, številka katastrske občine, številka stavbe, številka dela stavbe, ulica, hišna številka, poštna številka in druge. V tabeli Elektronske naprave so šifrirane posamezne naprave, iz katerih se izdaja račun, poleg povezave s poslovnim prostorom ima še polje šifra in naziv.

Račune, ki jih davčno potrjujemo, zapisujemo v vzporedno tabelo Račun davčnega potrjevanja, tam hranimo samo skupek podatkov, ki je potreben za davčno potrjevanje. Tabelo v podjetju uporabljamo v več produktih in nam je zagotovila bolj preprosto povezavo na naše standardne funkcije davčne blagajne.

Poleg tega potrebujemo še tabelo Parametri, Dnevnik sprememb poslovnega prostora in prenešene XML datoteke.

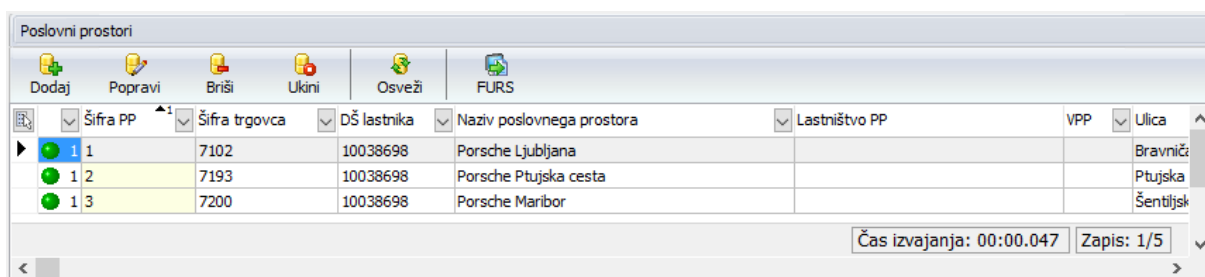
Povezavo med trgovcem in obveznikom fiskalizacije smo naredili na nivoju poslovnega prostora, saj trgovci v sistemu CROSS niso vedno pravni subjekti (Obveznik fiskalizacije), ampak ima lahko en pravni subjekt več trgovcev. Poenotili smo tudi uporabnika na računu s šifrantom Uporabniki, saj za potrebe davčnega potrjevanja potrebujemo uporabnikovo davčno številko. Ker davčne številke uporabnika ne moremo prejeti preko XML datoteke izdanih računov, je potrebno vsakega novega uporabnika sistema CROSS vnesti tudi v davčni blagajni, sicer se tak račun konča z napako in se ne potrdi.

4.3 Uporabljena programska orodja

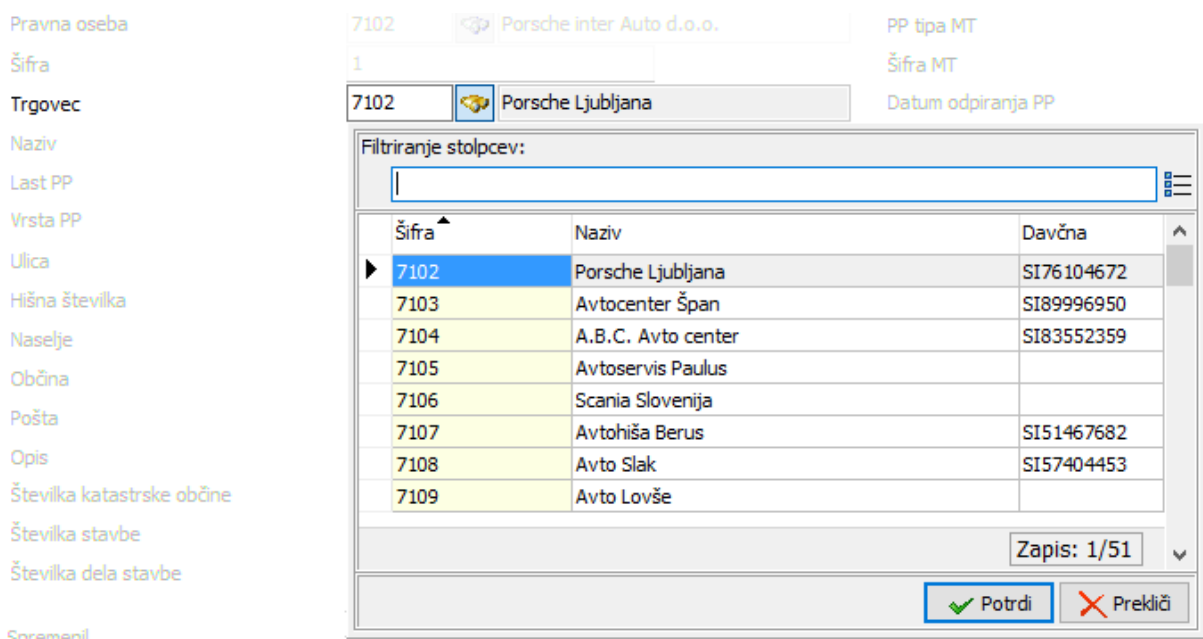
4.3.1 Razvojno okolje Delphi

Delphi [4] je programski jezik in programsko orodje za različne ciljne platforme. Programski jezik temelji na Pascalu, ki so ga so poimenovali kar objektni Pascal. Delphi je razvilo podjetje Borland leta 1995 kot orodje za hiter razvoj aplikacij, ki je nasledil Turbo Pascal. Prednost programskega orodja je dobra podpora podatkovnim bazam, močan nabor sistemskih in vizualnih knjižnic, hitro prevajanje [5] in vključen razhroščevalnik. Leta 2006 je Borland ekipo za razvoj preselil na hčerinsko družbo CodeGear [6], ki jo je leta 2008 kupilo podjetje Embarcadero Technologies. Leta 2015 je Embarcadero kupilo podjetje Idera [7], vendar so blagovno znamko Embarcadero ohranili.

V podjetju uporabljamo razvojno orodje Delphi XE3, to nam omogoča hiter razvoj programske opreme in boljše prilagajanje naročnikovim željam. Za ta namen imamo veliko lastno razvitih knjižnic (VCL), ki nam olajšajo hiter razvoj aplikacij in naše produkte naredijo uporabniku prijazne. Osnovni predstavitveni komponenti sta iDBGrid in iHelpEdit, prva omogoča pregled podatkov iz podatkovne baze, kot prikazuje slika 4.3, druga pa lažji izbor podatka pri urejanju podatkov, kot je prikazano na sliki 4.4. Poleg tega imamo še vrsto drugih uporabnih komponent za komunikacijo z bazo, olajšano gradnjo osnovnih oken in boljšo uporabniško izkušnjo.



Slika 4.3: Komponenta iDBGrid



Slika 4.4: Komponenta iHelpEdit

4.3.2 **Microsoft SQL server**

Microsoft SQL Server [8] je sistem za upravljanje relacijske baze podatkov, ki ga je razvil Microsoft. Njegova primarna naloga je shranjevanje in pridobivanje podatkov, ki jih zahteva drug program na istem računalniku ali preko mreže – vključno preko interneta. Glavni način pridobivanja podatkov s strežnika podatkovne baze SQL je s poizvedbami. Poizvedba je izražena po različici SQL, imenovani T-SQL, to je dialekt, ki si ga Microsoft SQL Server deli skupaj s Sybase SQL Server, na osnovi katerega je nastal. SQL Server vključuje optimizator, ki skuša poizvedbo optimizirati glede na ceno in sredstva, ki so mu trenutno na voljo. Pri vsaki poizvedbi optimizator gleda na sheme baze podatkov, statistike baze podatkov in trenutno obremenitev sistema. Nato se odloči, s katerim zaporedjem bo dostopal do tabele, navedene v poizvedbi, s katerim zaporedjem bo izvrševal ukaze in kakšen postopek dostopa bo uporabljal za dostop do tabele [9].

SQL Server omogoča tudi shranjene procedure. To so parametrizirane poizvedbe T-SQL, ki so shranjene na samem strežniku. Lahko sprejme vrednosti s strani odjemalca kot poslane vhodne parametre in pošlje nazaj rezultate kot izhodne parametre. Lahko kličejo preddefinirane funkcije in druge shranjene procedure, vključno z isto shranjeno proceduro (rekurzija) [10]. Shranjene procedure nam omogočajo, da določen del programske kode preselimo, da se izvaja na podatkovnem strežniku. To nam omogoča hitrejše izvajanje, ker po mreži ne prenašamo vmesnih poizvedb neke funkcije, ki bi tekla na aplikacijskem strežniku ali celo na odjemalčevi počasni internetni povezavi. Slabost razvoja s pomočjo shranjenih procedur je razhroščevanje, saj ni tako preprosto in pregledno kot v programskem orodju Delphi.

4.4 **Storitve aplikacijskega nivoja**

4.4.1 **iDataBroker**

Storitev, razvita v razvojnem okolju Delphi s pomočjo knjižnic REM objects, teče na srednjem aplikacijskem nivoju trinivojske arhiture. Glavni namen storitve je, da pri odjemalcu poenostavi in pohitri dostop do podatkovne baze. Celotna komunikacija poteka preko enih vrat na TCP/IP protokolu. Gonilniki ali dodatne DLL knjižnice pri odjemalcu niso potrebne. Celotna aplikacija na odjemalcu je samo ena izvršilna datoteka, zato je namestitev pri stranki zelo enostavna. Omogoča nam tudi kompresiranje podatkov, kar pomeni, da povezava deluje hitro in zanesljivo tudi preko počasnejših mobilnih povezav.

Storitev nam omogoča še nadzor nad napakami, ki se pojavljajo na odjemalcih, pregled aktivnih povezav, pregled najdaljših SQL poizvedb, omejevanje dostopa odjemalcem, kriptiranje povezave. Omogoča nam tudi nadzor in avtomatsko posodobitev verzij, ki tečejo na odjemalcih, tako smo lahko vedno prepričani, da ima odjemalec vedno najnovejšo različico.

4.4.2 iBroadcast

Gre za storitev, ki je bila namensko narejena za naročnika, na njej teče vsa pomembna poslovna logika – od uvoza XML datoteke do davčnega potrjevanja računa. Storitev preži na spremembe v določeni mapi in ko tam zazna novo datoteko, zanjo sproži uvoz. Vsak uvoz se izvede v ločeni niti. Ko se datoteka obdela, sledi še obvestilo odjemalcu, s katerim se avtomatsko sproži pogovorno okno za izpis računa.

Prvi korak je bil pravilen uvoz XML datoteke. Orodje Delphi nam je s pomočjo XML Data Binding Wizarda in XSD datotek močno olajšal uvoz. Čarovnik avtomatsko ustvari novo enoto, ki ima definirane vmesnike in razrede za vsa vozlišča v XML dokumentu. Poleg tega ustvari globalno funkcijo, v katero pošljemo XML dokument, ta pa nam vrne strukturiran objekt z vsemi podatki. Zataknilo se je samo pri dejstvu, da so bile posredovane XSD datoteke pomanjkljive in niso vsebovale vseh možnih polj, ki so nastopala pri dejanskih uvoženih XML datotekah. Tako smo morali določena polja in strukture dopolnjevati ročno, da smo zajeli vse pomembne informacije.

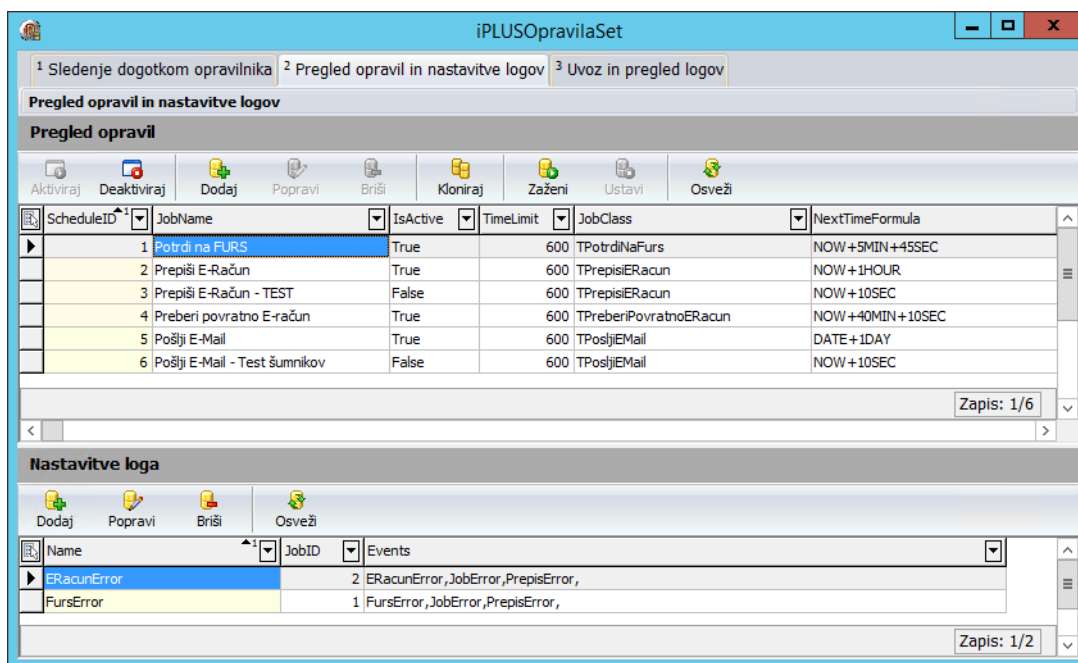
Ali je posamezen račun potrebno davčno potrditi, nam pove tip plačila. Če je označeno, da je plačilo z gotovino ali kartico, je račun nujno treba davčno potrditi. Nekateri trgovci so se odločili, da bodo davčno potrjevali vse račune. Oznako za to smo dali v šifrant poslovni prostori – potrjujemo vse razen internih računov. Z vsakim računom prejmemo informacijo o trgovcu, ta podatek nam pove poslovni prostor, blagajno razberemo iz številke računa. Uporabnik, ki je izdal račun, mora v našem sistemu imeti vpisano davčno številko, ker tega podatka žal nismo mogli prejeti iz uvožene XML datoteke oziroma ga v sistemu CROSS sploh ni mogoče voditi. Sledi kontrola vseh vhodnih podatkov za davčno potrjevanje, če je kakšen od podatkov napačen ali pa ga ni, ne moremo klicati funkcije za davčno potrjevanje. V tem primeru označimo račun s posebnim statusom, ki je uporabniku viden kot "napaka pred pošiljanjem" s podrobnim tekstom, za kakšno napako gre. V tem statusu izpis računa ni mogoč, treba je razrešiti problem ali s ponovnim pošiljanjem iz sistema CROSS ali pa s popravljanjem podatkov na odjemalcu davčne blagajne. V kolikor je kontrola podatkov uspešna, zberemo vse potrebne podatke in jih vpišemo v ločeno tabelo. Sledi izračun ZOI številke iz podatkov računa. Knjižnica SecureBlackbox podjetja Eldos nam omogoča, da

podatke elektronsko podpišemo z RSA-SHA256 algoritmom, saj Delphi v osnovi enkripticijskih algortimov nima. S temi podatki in s pomočjo MD5 zgoščevalne funkcije dobimo končno obliko ZOI. Sledi priprava XML dokumenta v predpisani strukturi po tehnični dokumentaciji, kateremu dodamo elektronski podpis, nato vzpostavimo varno TLS povezavo s certifikatom in podatke pošljemo na strežnik davčne uprave. Iz odgovora preberemo uspešnost davčnega potrjevanja, v kolikor je potrjevanje uspešno, preberemo identifikacijsko oznako računa (EOR) in jo vpišemo v tabelo računa za kasnejši izpis. Če pride do napake, postavimo status računa »v pošiljanju na FURS«, tak račun brez EOR kode se lahko izda stranki, ampak ga moramo v najkrajšem možnem času potrditi, to nalogo prevzame storitev iAgent, opisan v naslednjem poglavju.

Na koncu odjemalcu s prijavljenim enakim uporabniškim imenom, kot je na računu, pošljemo sporočilo za tisk računa ali obvestilo o napaki pri potrjevanju ali uvozu.

4.4.3 iAgent

Storitev, ki ji lahko dodajamo več nalog, vsaka naloga teče v svoji niti in ima možnost nastavitve svojega časovnika. Ena izmed glavnih nalog je ponoven poskus davčnega potrjevanja računa, ki v prvem poskusu ni uspelo. V praksi se to največkrat zgodi, ko strežnik davčne uprave v nekem momentu ni dostopen. Takrat dobimo napako o neodzivanju zaradi nastavljenih časovnih omejitev. Take primere potem ponovno poskušamo potrditi na vsakih 5 minut. V primeru ko v celim dnevu nismo uspeli potrditi določenega računa, nam naloga Pošlji e-mail vsako jutro pošlje obvestilo o napaki potrjevanja. Seznam prejemnikov je vpisan v nastavitvah, pregled in nadzor nalog pa opravljamo preko namenske aplikacije, opisane v naslednjem poglavju (iNadzor). Okno s pregledom in upravljanjem nalog je prikazano na sliki 4.5.

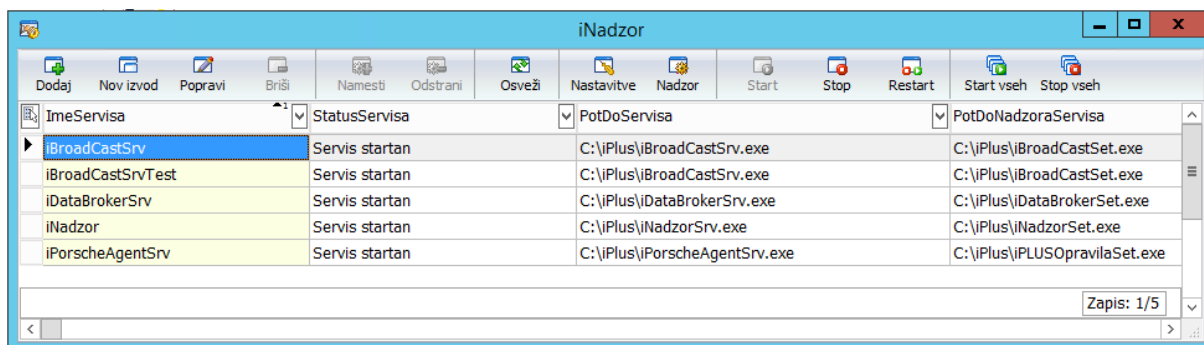


Slika 4.5: Pregled in upravljanje nalog iAgenta

Drugi sklop nalog zajema e-račune. Sestavlja ga dve nalogi, pošiljanje e-računov in branje povratne informacije. Uporabnik z odjemalcem pripravi in označi račun za pošiljanje, vsako uro se preveri, ali obstaja kakšen račun, ga shrani v stisnjeno mapo in pošlje na SFTP podjetja SETCCE. Povezavo preko SFTP vzpostavljamo s pomočjo knjižnice SecureBlackbox, saj Delphi XE3 ne podpira SFTP protokola. Ločena naloga bere povratne datoteke na SFTP, gre za XML dokumente z vsebino napake ali potrditvijo, da je bil prenos uspešen. Na vsakem računu vodimo tudi dnevnik pošiljanja, kjer je razvidno, kdaj smo račun poslali in kakšno povratno informacijo smo prejeli.

4.4.4 iNadzor

Gre za nadzorno aplikacijo, ki predstavlja uporabniški vmesnik za vse naše programske storitve. Omogoča nam namestitve, zagon, nastavitve in nadzor storitev. Uporabniški vmesnik s pregledom storitev je prikazan na sliki 4.6. Pri namestitvi gre za klasičen Windows ukaz namestitve storitve, ekvivalenten ukazni vrstici »/install«. Pri zagonu gre prav tako za splošen Windows ukaz zagona storitve. Za klic nastavitve in nadzora pa vsaka storitev uporablja svoj nastavitveni program. V nastavitvah so osnovne nastavitve, po vsaki spremembi je potreben nov zagon storitve. V nadzoru imamo vpogled v delujočo storitev, omogoča nam sledenje dogodkom (dnevnik) v živo, lepši pregled preteklih dnevniških datotek, omogoča nastavitve, specifične za storitev, za katere ni potreben ponoven zagon storitve.



Slika 4.6: Pregled in upravljanje storitev preko iNadzora

4.5 Odjemalec

Odjemalec predstavlja predstavitveni nivo trinivojske arhitekture. Omogoča pregled in tisk računov, pripravo e-računov ter pregled in urejanje šifrantov. Ob zagonu aplikacije se najprej pojavi prijavno okno, prikazano na sliki 4.7, v katerega uporabnik vpiše svoje uporabniško ime in geslo. Po prijavi uporabnik tipično nadaljuje delo v CROSS, saj ima naša aplikacija z uporabniškega vidika večino časa samo vlogo izpisa računa.

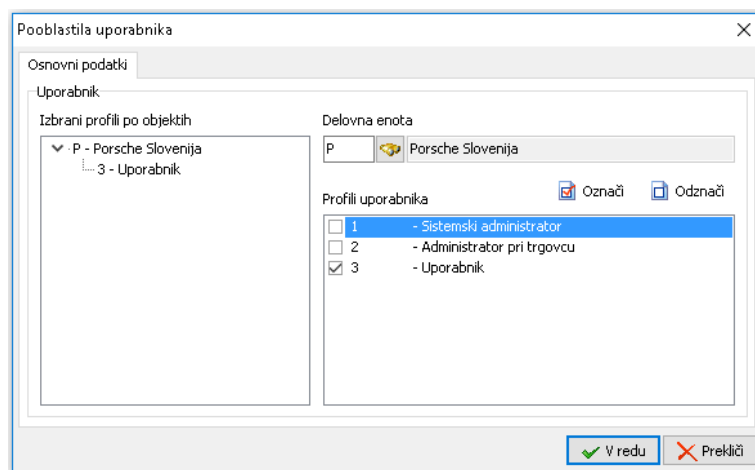


Slika 4.7: Prijavno okno

Izpis računa in zagotovitev podatkov zanj se je izkazalo kot eden večjih izzivov projekta, saj je bilo podatke, ki smo jih zajeli preko uvoza, potrebno pravilno interpretirati in jih ustrezno prikazovati na izpisu. Podpreti smo morali posebnosti, kot so račun na lizinško podjetje, na zavarovalnico, poseben izpis postavk paketnih storitev, spisek dodatne opreme vozila, predplačila. Potrebno je bilo veliko komunikacije z naročnikom in tudi kasneje v produkciji so se pojavljali primeri, ki jih nismo predvideli.

4.5.1 Uporabniki

Med osnovne šifrantе sodijo uporabniki. Pri odpiranju novega uporabnika je potrebno biti pozoren na šifro, saj ta mora biti enaka kot v CROSS, določiti je treba tudi šifro trgovca, pri katerem je uporabnik zaposlen, in davčno številko uporabnika, ki izdaja račune. Uporabniku določimo sklop pooblastil, ki jih definiramo v profilih. Primer izbire profila je prikazan na sliki 4.8.

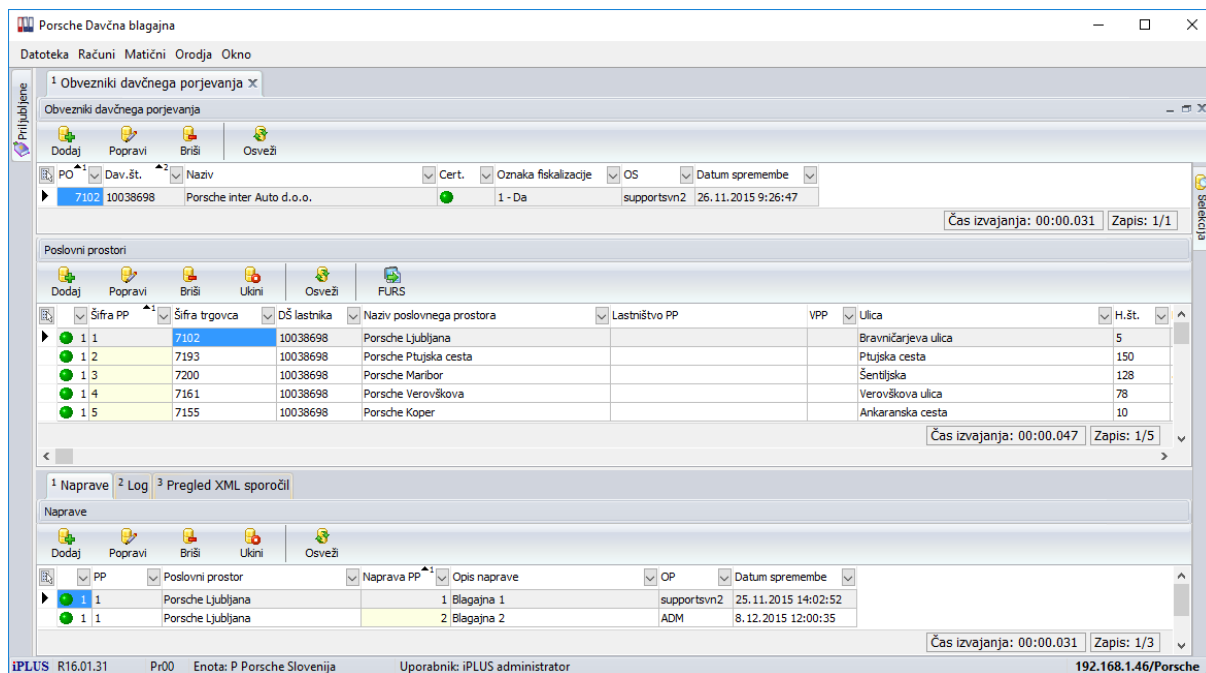


Slika 4.8: Izbira pooblastil uporabnika s pomočjo preddefiniranih profilov

Ob zagonu sistema nam je naročnik posredoval seznam vseh uporabnikov tako, da smo jih uvozili v sistem, vse kar je moral vsak uporabnik storiti je to, da je ob prijavi na novo določil geslo.

4.5.2 Nastavitve davčnega potrjevanja

Imamo osnovne nastavitve, ki so enotne za vse trgovce in pravne osebe, to so nastavitve naslova strežnika finančne uprave, kjer kličemo storitev, in nastavitve časovnika v primeru neodzivanja storitve. Bolj obsežne so nastavitve za vsako pravno osebo posebej, za ta namen imamo pregled s tremi preglednicami, na vrhu so obvezniki ali pravne osebe, pod njimi poslovni prostori in na dnu še elektronske naprave. Pregled teh nastavitvev, vključno s celotno programsko masko, je prikazan na sliki 4.9.



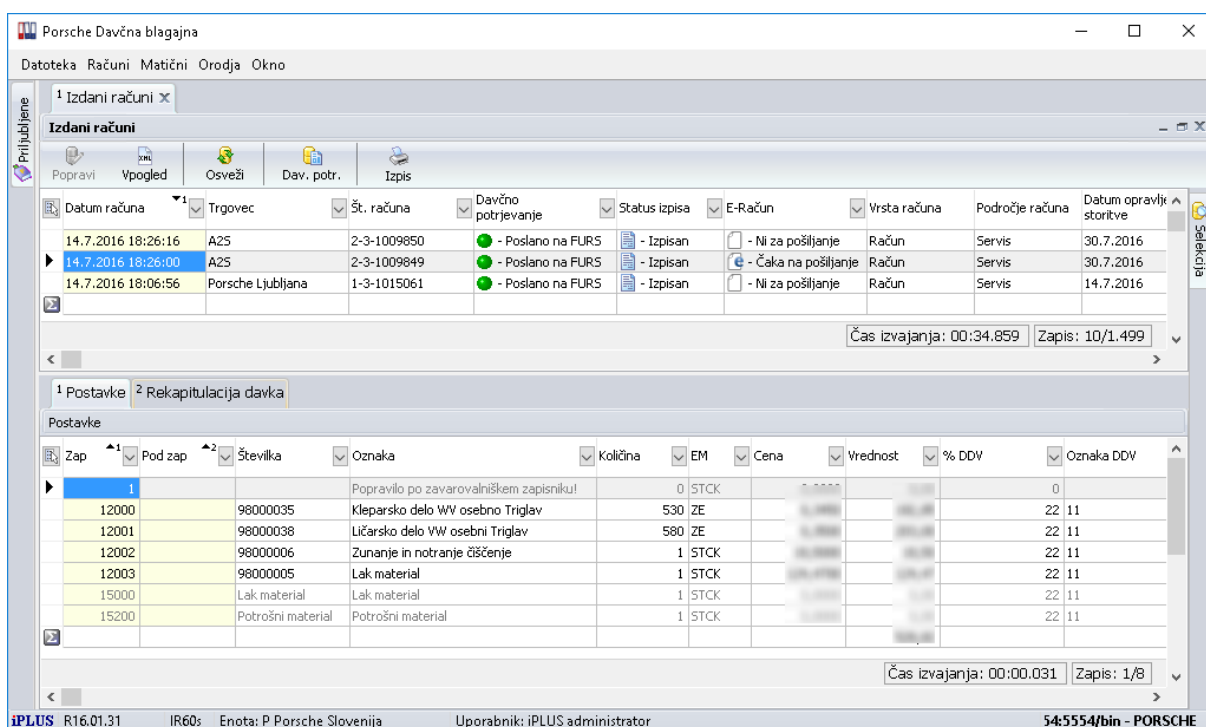
Slika 4.9: Pregled nastavitev in zgradbe aplikacije davčne blagajne

Vsakemu obvezniku moramo določiti naziv, davčno številko, certifikat in geslo za certifikat. Vsak obveznik ima enega ali več poslovnih prostorov, na poslovnem prostoru imamo veliko podatkov, ki jih zahteva davčna uprava in jih je treba poslati pri potrditvi poslovnega prostora na FURS. Vsa polja so vidna na sliki 4.10. V kolikor spreminjamo že potrjen poslovni prostor, ga moramo ponovno potrditi in spremembe zapisati v dnevnik. V spodnjem delu slike 4.9 so naprave, ki jim določimo številko in naziv. Spremembe ali dodajanje posamezne naprave ne vpliva na ponovno potrjevanje poslovnega prostora.

Slika 4.10 Nastavitve poslovnega prostora

4.5.3 Pregled računov

Okno za pregled računov predstavlja glavno okolje za uporabnika. Na njem uporabnik lahko tiska račune, pregleduje vse pomembne podatke računov, podrobnosti davčnega potrjevanja in uvožene XML datoteke. Glede na pooblastila, ki jih ima uporabnik, določimo, katere račune lahko pregleduje. Administrator pri trgovcu lahko s pomočjo selekcije izbere vse račune pravne osebe, uporabnik lahko pregleduje samo račune trgovca, administrator sistema pa nima omejitev pregleda. Omogočen je tudi ponoven izpis kopije računa, ki jo moramo številčiti. Okno s pregledom računov je prikazano na sliki 4.11.



Slika 4.11: Pregled izdanih računov

Za vsak račun, ki je davčno potrjen, omogočamo tudi pregled davčnih potrditev. Tu lahko vidimo skupek podatkov, ki je bil poslan na FURS, dnevnik davčnega potrjevanja in za vsak zapis v dnevniku poslano in prejeto XML sporočilo.

4.5.4 Pregled in urejanje e-računov

Pregled omogoča urejanje in pošiljanje e-računov preko podjetja SETCCE. Za vsak račun, ki ga uvozimo, se preveri prejemnika, ali ima oznako vladne organizacije. Če to oznako ima, potem račun označimo s posebnim statusom e-računa »čaka na pošiljanje«. Takemu račun lahko dodajamo dodatne priloge, ročno urejamo XML dokument in na koncu z gumbom »pošlji« tak dokument označimo za pošiljanje. Označene dokumente iz centralnega strežnika

pošlje storitev iAgent. V spodnjem pregledu vidimo dnevnik pošiljanja. Pregled je prikazan na sliki 4.12.

E-računi

Datum računa	Trgovec	Št. računa	Davčno potrjevanje	Status izpisa	E-Račun	Datum opravljene storitve	Datum valute	Preje
14.7.2016 15:44:09	Avtotohisa Vrtač	1-3-1011529	Poslano na FURS	Izpis	Potrjen			
14.7.2016 15:21:47	Avtotohisa Berus	1-3-1004022	Poslano na FURS	Izpis	Potrjen	6.7.2016		
14.7.2016 14:22:27	Avtotohisa Jurič	1-3-1002337	Poslano na FURS	Izpis	Čaka na pošiljanje	11.7.2016		

Čas izvajanja: 00:00.297 Zapis: 1/15

SETCCE Log

Datum	E-Račun	Koda	Opis	Status
14.7.2016 16:14:26	5 - Poslan		Poslano na SETCCE	
14.7.2016 16:15:33	8 - Potrjen	101	Izdaja racuna	0 - OK
14.7.2016 16:15:33	8 - Potrjen	208	UJP Net distribucija	0 - OK
14.7.2016 16:15:33	8 - Potrjen	406	UJP Net potrditev dostave	0 - OK

Čas izvajanja: 00:00.032 Zapis: 1/4

iPLUS R16.01.31 IR651 Enota: P Porsche Slovenija Uporabnik: iPLUS administrator 54:5554/bin - PORSCHE

Slika 4.12: Pregled e-računov

Poglavje 5 Sklepne ugotovitve

Izobraževanje uporabnikov in končno namestitvev pri trgovcih je izvedel naročnik, mi smo priskrbeli namestitveno datoteko s prednastavljenimi vrednostmi za povezavo z aplikacijskim strežnikom. Uporabnike takoj po namestitvi in zagonu aplikacije čaka prijavno okno, nobena druga nastavitvev s strani uporabnika ni potrebna. Pripomb končnih uporabnikov nismo prejeli, torej smo svoje delo opravili dobro. Z uvedbo davčne blagajne se naše delo pri naročniku ni končalo, še naprej sodelujemo pri manjših dopolnitvah in vzdrževanju sistema. Šele z večjim številom transakcij na podatkovni bazi so se pokazale določene pomanjkljivosti, ki v izvedbi niso bile predvidene. S postavitvijo primernih indeksov na najbolj prometnih tabelah smo se izognili nedostopnosti zaradi zaklenjenih zapisov v tabelah (deadlock). Sistem po tej optimizaciji deluje zanesljivo brez večjih izpadov. V pol leta je bilo prenešenih okrog 200.000 računov. Naročnik je vmes naročil še eno dopolnitev za pošiljanje e-računov, ki jih generirajo iz drugega sistema in jih ročno uvozijo preko odjemalca. To so računi, ki se glasijo direktno na uvoznika in ne na trgovca, davčno se ne potrjujejo, saj ne gre za gotovinske račune. Rešitev je bila potrebna zaradi lažjega urejanja in priprave XML datoteke za nadaljnji izvoz in pošiljanje e-računa.

Pri projektu smo spoznali veliko novega o varnih povezavah, digitalnem podpisovanju in obdelavi XML datotek s programskim orodjem Delphi. V starejših verzijah nekatere novejšje tehnologije niso bile podprte ali pa so delovale slabo, za take primere smo v preteklosti določene procese razvili z orodjem Microsoft Visual Studio. Tako mešanje dveh programskih orodij ni bilo najboljšje tako z vidika razvoja kot tudi kasnejšega vzdrževanja programske kode. Celoten razvoj s programskim orodjem Delphi je bistveno izboljšal preglednost in vzdrževanje programske kode. K temu so močno pripomogle tudi knjižnice SecureBlackBox podjetja Eldos, ki so nam omogočile digitalno podpisovanje in druge kriptografske funkcije. Vzpostavljen sistem knjižnic in orodij nam omogoča agilno razvoj vsebinsko zelo različnih projektov po individualnih željah naročnika.

Literatura

- [1] Davčne blagajne in vezane knjige računov. [Online]. Dosegljivo: http://www.fu.gov.si/nadzor/podrocja/davcne_blagajne_in_vezane_knjige_racunov_v_kr
- [2] Davčno potrjevanje računov – Tehnična dokumentacija. [Online]. Dosegljivo: <http://www.datoteke.fu.gov.si/dpr/files/TehnicnaDokumentacijaVer1.6.pdf>
- [3] Dealer Management System CROSS. [Online]. Dosegljivo: <http://www.porscheinformatik.com/loesungen/kfz-einzelhandel/dealer-management-system-cross.html>
- [4] Online Help for Delphi® XE3 and C++Builder® XE3. [Online]. Dosegljivo: http://docwiki.embarcadero.com/RADStudio/XE3/en/Main_Page
- [5] Performance Comparison from Delphi 2010 to XE6. [Online]. Dosegljivo: <http://blogs.riversoftavg.com/index.php/2014/05/12/performance-comparison-from-delphi-2010-to-xe6-part-2/>
- [6] Borland forming CodeGear to focus exclusively on developer productivity. [Online]. Dosegljivo: <http://edn.embarcadero.com/article/33819>
- [7] IDERA Announces Intent to Acquire Embarcadero. [Online]. Dosegljivo: <https://www.idera.com/about/news/pressreleases/2015/idera-announces-intent-to-acquire-embarcadero>
- [8] Microsoft SQL Server [Online]. Dosegljivo: [https://msdn.microsoft.com/en-us/library/mt590198\(v=sql.1\).aspx](https://msdn.microsoft.com/en-us/library/mt590198(v=sql.1).aspx)
- [9] SQL Statement Processing [Online]. Dosegljivo: <https://msdn.microsoft.com/en-us/library/ms190623.aspx?f=255&MSPPError=-2147217396>
- [10] Stored Procedure Basics [Online]. Dosegljivo: <https://msdn.microsoft.com/en-us/library/ms191436.aspx>